ISSN 0112-3351

BEEBLET

CONTENTS

Editorial 2
WAVEGEN 3
MicrCODIL (Advert) 4
Extending Sound to a Philips CF1114 Monitor 5
Classifieds (For Sale) 5 and 27

*FX Calls III

  Flipping Cursors and Tabs 6
Hint for Function Key Users 9
Anyone for Tennis? 10
Remember Melenie? 16

Beebug FILER

  Updated Sorting Routine 18
  Filer Notes 23
Glossary of Terms and Quotes 24
Programming Red Keys (Enquiry) 26
Database Services (Advert) 27
Classifieds (For Sale) 5 and 27
Membership 27
Local User Group Meetings, Contacts 28
Beeblet - Contributions and Deadlines 28

Volume 6 Number 4 June 1988

BBC / ACORN

COMPUTER USER GROUP NZ (inc)

EDITORIAL

With fewer contributions, this issue is a little briefer than I like to see each month. However, on your behalf, I would like to thank the following for their assistance:

Anne Flower	Wellington
Don MacDonald	Wellington
Rene Turner	Hamilton
Mike Shores	Auckland

Winter is setting in; the weather is a bit cooler and the evenings longer. Now is the time to finalise that program you were writing and forwarding it for publication. Or maybe you have just bought hardware or software which you could review for the benefit of other readers.

I cannot publish what I don't receive. So if you perceive a deficiency, be positive and see what you can do about it.

The July issue should contain the next in Anne Flower's excellent series on *FX calls, the programming of Function keys, and the results of Bernie Kay's experimentation with the extension of sound to a CF1114 Monitor. As for the rest of that issue - that is up to you.

Happy computing.

Graham Ellett

NOTICE TO LOCAL USER GROUP COMMITTEES:

Now that your AGMs are behind you, please check the accuracy of the information listed on the back page relating to meetings and local contacts in your area. Advise the Editor of any corrections required.

::

DISCLAIMER

The views of reviewers, contributors and/or Editorial committee are not necessarily shared by

THE BBC/ACORN COMPUTER USER GROUP NZ INC.

::

WAVEGEN

by Mike Shores, Auckland.

This short program is published without further explanation. Readers are invited to examine its purpose and method of operation for themselves.

```
 10 REM ****************************
 20 REM *     WAVEGEN program by    *
 30 REM *     SHORESOFT (c) 1987    *
 40 REM ****************************
 50 REM Program originally concerned with setting up
 60 REM Nanocomp(W.W. Jan.81) to produce exact
 70 REM frequencies.  However the calculation
 80 REM is valid on a general basis.
 90 *TV0,1
100 MODE0:VDU14,19,1,2;0;0;0;
110 Y=350:K=200
120 PROCsquarewave:PROCcycletime
130 VDU28,2,11,70,1
140 REPEAT
150 CLS:INPUT''"F= ";F
160 PRINT'"T= ";10^6/F;"microseconds"
170 PRINT''"So the timing periods between toggling= ";
    10^6/(F*2);"musecs"
180 PRINT'TAB(25)"P R E S S    S P A C E"
190 REPEATUNTIL GET=32
200 UNTIL FALSE
210 END
220
230 DEFPROCsquarewave
240 X=0:MOVE X,Y
250 REPEAT
260 DRAW X,Y+K:DRAW X+K,Y+K:DRAW X+K,Y:DRAW X+2*K,Y
270 X=X+2*K
280 UNTIL X>1000
290 ENDPROC
300
310 DEFPROCcycletime
320 X=0:REPEAT:X=X+2*K:UNTIL X>500:X=X-4*K
330 MOVE X+2*K,Y-20:        DRAW X+2*K,Y-100
340 MOVE X+5+2*K,Y-50:      VDU5:PRINT"<":VDU4
350 MOVE X+4*K,Y-20:        DRAW X+4*K,Y-100
360 MOVE X+(4*K)-15,Y-50:   VDU5:PRINT">":VDU4
370 MOVE X+20+2*K,Y-60:     PLOT21,X-15+4*K,Y-60
380 MOVE X-10+(3*K),Y-50:   VDU5:PRINT"T":VDU4
390 ENDPROC
```

MicroCODIL

In the mailbag this month we have received brochures from CODIL Language Systems.

What is MicroCODIL?

MicroCodil is a novel computer language system which concentrates the results of many years university research into a BBC computer. It combines detailed observations as to how people handle information with the latest ideas in the fields of educational technology, human-computer interaction and knowledge based systems.

It uses colour, windows, menus, help and diagnostic facilities to form a system which is inherently simple for novices to use. In the classroom MicroCODIL can play a number of distinct roles. Its flexible approach to the storage and retrieval of information makes it ideal as a data base language for subjects as diverse as biology, history, chemistry and business studies.

At a secondary school level MicroCODIL allows pupils to play with information in a way which has much in common with Papert's use of Logo to encourage primary school children to conceptualise geometry. As such it is an ideal language for encouraging imagination and experimentation in computer appreciation and information technology classes.

MicroCODIL comes complete with a dozen ready made teaching packages with interesting names such as BOGBOD, CRISIS, INVENT, PRICE, QUEST, MUSIC, SOLIDS, etc. It is designed to run on virtually any BBC based system from Model B with a single sided 40 track drive to am Archimedes in emulator mode. It is available in DFS or ADFS to suit.

If you would like further details, send a stamped self addressed envelope to: The Secretary, BBC/Acorn User Group NZ, Inc., PO Box 9592, Wellington, for a copy of the brochures (which include costs) or direct to:

Chris Reynolds
33 Buckingham Rd
Tring, Herts, HP23 4HG
ENGLAND

EXTENDING SOUND TO A PHILIPS CF1114 MONITOR

Tony Wattam of Taradale writes in requesting advice on using the TV speaker of a Philips 14" CF1114 monitor for the Beeb's sound. You might remember that Paul Collins reviewed the use of the monitor in the October 1986 issue of *Beeblet*.

Tony notes that there is an audio-in socket, but in Paul's diagram of the plugs, the centre pin of the TVs RGB socket is shown as an audio-in. Does this mean that with a suitable modification to the Beeb, a single cable will do for both RGB and sound, or do we have to have a separate cable for audio? The ideal would be to have the option of the TV or Beeb's speaker - or is this unnecessarily complicated?

Editorial Comment:

The CF1114 appears to have become popular amongst home computer users. With the demise of an old TV, and following Paul's article a CF1114 is now being used at the editor's desk. It was a good choice.

I have spoken to Paul Collins. He admits having taken the easy way out and has run a separate lead for the sound. This method has the advantage of requiring no modifications to the Beeb - the centre pin of the RGB socket on the Beeb normally has +5 volts on it.

However, I have discovered that fellow committee member, Bernie Kay, is in the process of carrying out the modification on his own Beeb and it should be possible to publish details next month.

For Sale:

1 x INTERWORD 32K wordprocessor ROM, new, complete with manuals, etc.,	$120.00
1 x INTERBASE ROM database, new, complete with manuals, etc.,	$150.00

Contact: Barry Borman, 12 Auckland Tce, Melrose,
Wellington. Tel: (04) 898 855

*FX CALLS III

FLIPPING CURSORS AND TABS

by Anne Flower, Wellington

After last month you may suspect the reason you are more likely to see hawks than eagles in NZ. If you were getting worried about your typing, we'll spend some time looking at *FX calls relating to the keyboard.

From any issue of any computer magazine:

> "Some of the keys on my keyboard have suddenly stopped working..."

The long-term solution is to go hunting for cracks in tracks and filling them in. The short-term solution (ie., while the soldering iron warms up and you frisk the kids' pockets for a magnifying glass) is to redefine some other key to print the missing character. Sure, you can define the red keys, but you may have other uses for most of those. How about the TAB key - what if whenever you pressed TAB you got that missing "e"?

If you were to type in

```
PRINT ASC"E"
```

the computer would say "69" was the ASCII value of "E".

The ASCII value of TAB is 9 normally, but *FX219 will allow you to assign any other value to it. (Any other?). The BBC User Guide p.497, and Master Welcome p.191 have a picture of the keyboard showing the normal ASCII values of each key. So to cope with that E key not working, try:

```
*FX219,69
```

Tried it yet? Did you find, like I, that if caps or shift lock was ON, you got "e", and with them both OFF, you got "E"!! Round the other way from the rest of the keyboard isn't it? Perhaps we should really redefine TAB as "e" whose ASCII value is 101:

```
*FX219,101
```

Try it. Just like any other self-respecting key isn't it?

You can also redefine TAB to be one of the red function keys, by giving it a new ASCII value in the range 128-143

eg.	*FX219,128	makes it behave like f0
	*FX219,130	like f2
	*FX219,138	like f10 which is also BREAK - but you won't get the effects of Break when you use TAB as *KEY10!!

MORE FUNCTION KEYS

Although the computer won't allocate any more RAM for your key definitions, you still may have need for more than nine red keys + Break with lots of short definitions.

*FX4,2 makes the COPY and arrow (cursor) keys behave like function keys - that is, you can define them with *KEY command, and get them working that way. When you want to go back to using the cursor and copy keys as 'normal', *FX4,0 will reset them.

If, like me, you can't afford to have valuable brain cells tied up remembering which way round they go, or you are a lazy typist, can you spare one of the normal function keys to swap between the two uses of the editing keys? Something along the lines of the PROC we had to resort to back in Part I because the value in the X register has to keep being changed:

```
*KEY3IFX%=2X%=0ELSEX%=2¦MA%=4:CALL&FFF4¦M
```

looks promising - and it works.

Press f3 and try the cursor keys - give them some definitions to be sure. It works - until you use X% or A% in some other way that changes the value - or you press that key during a program and corrupt its A% or X% values.

That's because, unlike PROC parameters, variables aren't LOCAL to a key definition. It isn't as if you could put the whole DEFPROC into a key and use it. And even if you could, the whole idea was to get as many different key definitions as possible, not use up all the available memory for them in one giant definition. I consulted my other half (remember *FX254,0?) most unfairly - ie before 11am on a Saturday.

Store the values in some spare bit (byte?) of memory - say in zero page where there are odd bytes available for that sort of thing, and reinstate them when needed... and use OSCLI... and wasn't there an *FX call that did this?

"*FX1 Reserved for Applications programs" says BBC User Guide.

"Read/write the user flag" says the Advanced User Guide.

"This call provides a means of writing data to the user flag (location &281). The flag is zero by default" says the Master Reference Manual I.

Haven't Acorn improved in 7 years!!

So what this means is that we can place zeroes and twos alternately into this byte of memory. Why choose this byte and not one in zero page? Mainly because it's less likely to be used by any program you're using (I hope - statistics please).

The arithmetic expression to alternate between 2 and 0 in this byte is:

```
?&281 = (?&281=2)*2+2
```

In English it is approximately translated as:

"The value in location &281 is equal to" and "Tell me if it is true that the value in location &281 equals 2 by calling this -1, or else 0 if it isn't the same. Multiply this response by 2 and add 2 to it" The resultant number is what will now be in location &281. The rest of the function key definition is another way of putting a variable into an *FX call (assuming not BASIC I):

```
*KEY3?&281=(?&281=2)*2+2:OSCLI"FX4,"+STR$?&281|M
```

The manuals also refer to *FX241. It not only reads and writes to &281, but also can toggle between two values if you ask it to. I haven't understood it enough to re-explain it - someone else? - but take my word for it, that the following definition works. It has the advantage of being shorter still, and retains that advantage of no variable names in sight:

```
*KEY3*FX241,2,2|MOSCLI"FX4,"+STR$?&281|M
```

All this just goes to show how involved these un- semi-documented *FX calls can get, how easily they lead you away from the original demand made by lazy typists, and into the realms of "perfection at all costs". You haven't seen the very first *KEY3 definition I wrote either!

I haven't touched *FX4,1 for now, and would someone else please explain *FX221-4, what they are and when they would get used.

THAT SINKING FEELING

Last month I showed you how to sink a printer - or at least let out its bunghole to allow data to drain away. What about those infernally noisy programs?

```
10¦PRINT"Sinker program":VDU7:wait=INKEY(100)
20¦PRINT"Will *FX210 help?":*FX210,1
30¦VDU7:wait=INKEY(100)
40¦PRINT"How long will it last?":*FX210,0
50¦*FX214,255
60¦VDU7
70¦END
```

Unfortunately the effect only lasts until Break is pressed, so redefine that as

```
*KEY10*FX210,1¦M
```

But if the kids are too smart and do a CTRL-Break you'll just have to disconnect the speaker. If they are even smarter than that, well that's hereditary isn't it?

::

HINT
for
FUNCTION KEY USERS

To eliminate the unfortunate habit the User Function Keys have of printing their instructions on the screen: insert CTRL L (¦L) after each command. This clears the text area before the command is executed.

```
eg:    *KEY0 OLD¦L¦MLIST¦L¦N¦M
```

David Kenealy, Wellington.

CHAMPIONSHIP TENNIS

by D S McDonald, Wellington.

Program "TENNIS" plays Championship Tennis - 5 setters without tie-breaks, like Wimbledon Men's Singles. It plays N (up to 200) matches between 2 players.

The first server is determined by the toss of a coin and service alternates after every game thereafter. The choice of ends is elected by the receiver and the players change ends after every odd game of the competition.

The scoreboard shows matches won, sets won in current match, games in each set, total games won by each player, and total points won. See Figure 1. It also shows point by point game score in the first game and the current game, the winner of the last game, and clocktime starting from the beginning of the competition.

The input to the program includes :

1. Players' names and service percentages (how many points out of 1000 they should win on average respectively when serving from either end, "UP" or "DOWN"),
2. number of matches in competition, and
3. a random number whereby the whole competition can be duplicated or played again by chance.

The scoreboard can be frozen at the end of a match for up to 1 hour by pressing the space bar in real time. The scoreboard is restarted by pressing another key.

The speed of play can be retarded by up to 1 second per point by pressing 0-8 in real-time (initially set at 2.) The maximum speed is 13 points played per second. The current speed is shown as "SPDn", for example.

At the end of the competition the player can obtain an analysis of games won by attacking/defending player and length of game. Press A (Aye) for analysis. This erases the scoreboard. Finally, the operator may play again (Y) or quit (N).

A small overall points advantage should result on average in a slightly greater games and sets advantage. The overall winner after a series of matches should usually be the player who starts with a theoretical service points superiority.

There are 4 components of a player's expected performance, viz. the service percentages. Thus by experiment, an expert player can determine which part of his game, serving up or down or defending, would most benefit from improvement.

```
 10 PRINT"PROG. TENNIS, 5.3.88,  PRESS ANY KEY"
 20 PRINT"OR FUNC KEY f7 FOR EXAMPLE"
 30 *KEY7XLEWIS|MLENDLE|M499|M465|M525|M549|M9|M55|M
 40 A$=GET$
 50 DIMPLAYER$(2),WIN%(2,2),DEN$(2),POYNTS(2)
 60 DIMGAME(2,5),SET(2),MATCH(2),A%(2,50)
 70 DIMNG(2),NPT(2),AT%(2)
 80 MODE7
 90 PROCTITLEINTRO
100 FORM=1TOMATCHES
110 PROCMATCH
120 NEXTM
130 *FX21,0
140 PRINTTAB(0,23)"PRESS A FOR ANALYSIS OF GAMES, OR SPC
    BAR";
150 A$=GET$
160 IFA$="A"THENPROCANALYSIS
170 *FX21,0
180 PRINTTAB(0,23)"DO YOU WANT TO PLAY AGAIN, Y/N.?";
    SPC(7);
190 A$=GET$
200 IFA$="Y"THEN80
210 IFA$="N"THENSTOP
220 GOTO180
230 END
240 DEFPROCTITLEINTRO
250 FORL=1TO2
260 VDU129,157,132,141:
    PRINT"CHAMPIONSHIP (WIMBLEDON) TENNIS."
270 NEXTL
280 PRINT"BY D.MC DONALD, 5.3.1988"
290 PRINT"ENTER PLAYERS NAMES AND EXPECTED NO. OF POINTS
    EACH PLAYER WINS PER 1000 SERVES FROM EACH END."
300 PRINTTAB(11,6);"ENDS",TAB(10,7);"UP | DOWN"
```

..... *TENNIS* Contd >>>

..... *TENNIS* Contd >>>

```
310 PRINTTAB(0,8);"SERVER 1";TAB(18,8);"PLAYER 1"
320 PRINTTAB(0,9);"SERVER 2";TAB(18,9);"PLAYER 2"
330 FORP=1TO2
340 PRINTTAB(27,P+7);:VDU 128+P:INPUTPLAYER$(P)
350 NEXTP
360 FORP=1TO2
370 FORE=1TO2
380 REPEAT
390 PRINTTAB(6+4*E,P+7);"X":INPUTTAB(6+4*E,10);WIN%(P,E)
400 UNTIL(WIN%(P,E)>0)AND(WIN%(P,E)<1000)
410 PRINTTAB(5+4*E,P+7);:VDU128+P:PRINT;WIN%(P,E)
420 NEXTE
430 NEXTP
440 REPEATINPUT"HOW MANY MATCHES TO PLAY 1-200",MATCHES
450 UNTIL(MATCHES>0)AND(MATCHES<=200)
460 INPUT"ENTER RND NO., (MAX 6 DIGITS)",R
470 IFR<2THEN460
480 X=LOGR
490 X=RND(-2E9*(X-INTX))
500 SERVER=RND(2)
510 FSERV=SERVER
520 REM FIRST SERVER RANDOM DEPENDING ON TOSS OF COIN , R
530 IFWIN%(SERVER,1)<WIN%(SERVER,2)THENE=1 ELSE E=2
540 DEN$(1)="UP":DEN$(2)="DOWN":REM DEN = ENDS
550 PRINT"COMPETITION OF ";MATCHES;" MATCHES."
560 FORP=1TO2
570 MATCH(P)=0:NG(P)=0:NPT(P)=0
580 FORPT%=0TO50:A%(P,PT%)=0:NEXTPT%
590 NEXTP
600 TX=TIME
610 A$="2" : REM MED SLOW
620 ENDPROC
630 DEFPROCMATCH
640 FORP=1TO2:SET(P)=0
650 FORS=1TO5:GAME(P,S)=0
660 NEXTS
670 NEXTP
680 S%=0
690 REPEATS%=S%+1
700 PROCSET
710 UNTILSET(POYNT%)=3
720 MATCH(POYNT%)=MATCH(POYNT%)+1
730 VDU7
740 PROCSCORE
750 B$=INKEY$(250)
```

..... *TENNIS* Contd >>>

..... *TENNIS* Contd >>

```
 760 IF(A$="9")OR(A$=" ")THENA$=INKEY$(360000)
 770 ENDPROC
 780 DEFPROCSET
 790 G%=0
 800 REPEATG%=G%+1
 810 PROCGAME
 820 UNTIL(GAME(POYNT%,S%)>5)AND(GAME(POYNT%,S%)-
     GAME(3-POYNT%,S%)>1)
 830 SET(POYNT%)=SET(POYNT%)+1
 840 ENDPROC
 850 DEFPROCGAME
 860 PRINT"SERVER";:VDU128+SERVER:
     PRINTPLAYER$(SERVER);" FROM ";DEN$(E);" END.";
 870 FORP=1TO2:POYNTS(P)=0:NEXTP
 880 REPEAT
 890 IFRND(1)*1E3<WIN%(SERVER,E)THENPOYNT%=SERVER
     ELSE POYNT%=3-SERVER
 900 POYNTS(POYNT%)=POYNTS(POYNT%)+1
 910 NPT(POYNT%)=NPT(POYNT%)+1
 920 PRINT;POYNTS(1);"-"POYNTS(2)",";
 930 B$=INKEY$(10*VAL(A$))
 940 IFNOT(B$="")THENA$=B$
 950 UNTIL(POYNTS(POYNT%)>3)AND(POYNTS(POYNT%)-
     POYNTS(3-POYNT%)>1)
 960 PT%=POYNTS(3-POYNT%):REM LOSERS GAME POINTS
 970 VDU128+POYNT%:PRINT"GAME";PLAYER$(POYNT%);:
     VDU135:PRINT;"("TIME-TX")";SPC(40)
 980 GAME(POYNT%,S%)=GAME(POYNT%,S%)+1
 990 NG(POYNT%)=NG(POYNT%)+1
1000 IFWIN%(SERVER,E)>500THENATKR%=SERVER
     ELSEATKR%=3-SERVER
1010 IFPOYNT%=ATKR%THENA%(1,PT%)=A%(1,PT%)+1
     ELSEA%(2,PT%)=A%(2,PT%)+1
1020 SERVER=3-SERVER
1030 IFSERVER=FSERV THENE=3-E
1040 PROCSCORE
1050 ENDPROC
1060 DEFPROCSCORE
1070 PRINTTAB(0,16);"MATCH ¦ SETS ¦ GAME SPD"A$" ¦ NG  ¦
     NPTS"
1080 FORP=1TO2
1090 VDU128+P:PRINT;MATCH(P);TAB(8)SET(P)TAB(13);:FOR
     S=1TO5:PRINT;GAME(P,S)" ";:NEXTS:PRINTTAB(27);NG(P)"
     "NPT(P)
1100 NEXTP
```

..... *TENNIS* Contd >>

..... *TENNIS* Contd >>>

```
1110 PRINT"FAST OA-Z ¦ SLOW 1-8 ¦ FREEZE 9,SPC"
1120 ENDPROC
1130 DEFPROCANALYSIS
1140 CLS:PRINT"CHAMPIONSHIP TENNIS."''"ANALYSIS OF GAMES
     WON BY ATTACKING AND  DEFENDING PLAYERS BY LENGTH OF
     GAME."'"LENGTH OF GAME MEASURED BY LOSERS GAME
     POINTS."
1150 PRINT"LNGTH/GAME ¦ ATTACKER GAMES ¦ DEFENDER"
1160 FORPT%=0TO50
1170 IFA%(1,PT%)+A%(2,PT%)>0THENPRINTPT%,A%(1,PT%),
     A%(2,PT%)
1180 NEXTPT%
```

..... *TENNIS* Contd >>>

Figure 1: Scoreboard panel of Program TENNIS.

```
           CHAMPIONSHIP (WIMBLEDON) TENNIS
           BY D.MCDONALD, 5.3.1988
           ENTER PLAYERS NAMES AND EXPECTED NO. OF
           POINTS EACH PLAYER WINS PER 1000 SERVES
           FROM EITHER END.
                         ENDS
                       UP  DOWN
           SERVER 1   499  465   PLAYER 1?LEWIS
           SERVER 2   525  550   PLAYER 2?LENDLE
           HOW MANY MATCHES TO PLAY 1-200? 10
           ENTER RANDOM NUMBER (MAX. 6 DIGITS)? 55
           COMPETITION OF 10 MATCHES.
1st Game:  SERVER LENDLE FROM UP END. 0-1,1-1,2-1,
           3-1,4-1,GAME LEWIS (25)

           MATCH ¦ SETS ¦ GAMES  SPDO   ¦ NG ¦ NPTS
           3       0      5 4 1 0 0      157  1162
           7       3      7 6 6 0 0      204  1275
           FAST OA-Z ¦ SLOW 1-8 ¦ FREEZE 9 SPACE
Last Game: SERVER LENDLE FROM UP END. 0-1,1-1,1-2,
           1-3,1-4,GAME LENDLE (18452)
           PRESS A FOR ANALYSIS OF GAMES, OR SPC BAR
           DO YOU WANT TO PLAY AGAIN, Y/N.?
```

Notes: Number in brackets after player's name is time in centiseconds since the beginning of play.

The player who cannot win most of his own service points (LEWIS) in this case nevertheless won 30% of matches.

..... *TENNIS* Contd >>>

```
1190 PRINT"TOTAL::::";
1200 FORP%=1TO2
1210 AT%(P%)=0
1220 FORPT%=0TO50
1230 AT%(P%)=AT%(P%)+A%(P%,PT%)
1240 NEXTPT%
1250 PRINTAT%(P%);
1260 NEXTP%:PRINTAT%(1)+AT%(2)
1270 PRINT"PRESS SPACE TO CONTINUE OR"'" A-Z TO REVIEW
     SCOREBOARD ?";
1280 A$=GET$
1290 IFA$<>" "THENPROCSCORE
1300 ENDPROC
```

Figure 2: Analysis Panel of Program TENNIS.

```
CHAMPIONSHIP TENNIS.

ANALYSIS OF GAMES WON BY ATTACKING AND
DEFENDING PLAYERS BY LENGTH OF GAME.
LENGTH OF GAME MEASURED BY LOSERS GAME
POINTS.

LNGTH/GAME ! ATTACKER GAMES ! DEFENDER
0            21               16
1            48               39
2            63               54
3            41               24
4            17               13
5            9                7
6            2                3
7            0                1
8            3                0
TOTAL:::   204               159      361

PRESS A-Z TO REVIEW SCOREBOARD

DO YOU WANT TO PLAY AGAIN, Y/N.?
```

Note: LENDLE was regarded as attacking player even when LEWIS served because Lewis was not expected to win the majority of his service points. See service percentages.

REMEMBER MELENIE?

A progress report from

Rene Turner, Hamilton

Remember Melenie; the little girl with the high spinal injury in our hospital? Well, I've been asked to write a progress report since she has been provided with a Master 128 computer and a whole range of software.

Initially Melenie was thrilled to have a clown on a television screen talk just to her, and this provided enough stimulation for her to expend the great effort which she required to move her hand to touch a "touch sensitive switch" in order to have the computer draw in segments of a picture. These very simple beginnings have led to a small team being established to encourage and educate Melenie as well as provide her with entertainment.

Each week, she has Noeleen from the Crippled Childrens' Society visit to educate her at the hospital. She has progressed from the single switch to banks of four large switches which she can now operate with both hands and both feet. Don't get me wrong, it's not easy; each movement is slow and takes a great deal of effort but every day there is a new minor improvement.

Twice each week, Melenie is allowed out for the afternoon to Kindergarten where she mixes with other children and enthrals them with her computer.

Melenie's father visits every day and they have a weekly routine of time with the computer (I hear Melenie is teaching Dad how to use it); time for them to go for walks in the park by the lake (Melenie in her buggy), and they sometimes go home for the night, to McDonalds for treats, etc, etc.

Life is certainly looking up - the times that she is off the respirator get a little longer each day, she has been able to move herself along with her toes, when she is placed in a special framed chair on wheels that holds her standing upright.

No great momentous miracles, just hard work, and every now and then a minor miracle depending on how you look at it.

The most notable has been that a little girl who said "I can't" and never tried has now found the will and the desire to reach out and learn that she CAN if she really wants to.

Melenie is now a happy outgoing adventurer learning more of what she can do each day.

Oh, by the way, the Accident Compensation Commission don't think Melenie is old enough to benefit from a computer, and when she is, an APPLE will be more suitable but not before she is 12 years old at a minimum. I think we've proved them wrong on all counts; I just wish we could find a way to convince them.

Thank you all for your help and software.

....

Ed: Thanks for the update, Rene. Its great to hear that Melenie is making such great progress. Incidentally, Melenie will now be four years old.

We first told you about Melenie in the August 1987 issue of Beeblet *(Vol 5 No 7 Page 39). At the time we requested software and/or hardware assistance. This is a continuing project and I am sure Rene will be pleased to hear from anyone who can provide further assistance.*

Rene's address is: 28 Prisk St
Hamilton
Telephone (071) 435 779

or: C/o The Anaesthetic Department
Waikato Hospital

IMPROVED SORTING ROUTINE

FOR

BEEBUG FILER

by Jan Stuurman.

FILER was published in *Beeblet* in a series of issues starting with Vol 4 No 8 and concluding in Vol 5 No 5. *Beeblet* acknowledges with gratitude the permission granted by *Beebug* magazine (which originally published the series) to publish the series of articles on *Beebug* FILER.

It has become clear that there has been at least two updates on the sorting routine originally published in *Beeblet* in May 1987. Because there has been more than one update, it has been decided to publish the latest version of FILERS in full.

Readers are invited to compare their version of FILERS with what follows and make updates as required. The background information published previously still applies.

NOTE: Some contributors to *Beeblet* have requested that the FILER series of programs be copied on to their discs before they are returned. This offer still stands and the FILERS update will be supplied if required before returning a contributor's disc. Note, however, that there may be some delay in returning such discs - publication of *Beeblet* must take priority.

```
 10 REM Program FILERS
 20 REM Version B2.1
 30 REM Author Jan Stuurman
 40 REM BEEBUG March 1987
 50 REM Program subject to copyright
 60 :
100 MODE135:HIMEM=PAGE+&1800:Z%=0
110 ONERRORPROCerror:END
120 PROCsetup
130 PROCtitle
140 PROCinput
```

....*FILERS* V B2.1 Contd>>

....*FILERS* V B2.1 Contd>>

```
 150 PROCsort
 160 PROCclose
 165 PRINT"Press any key for main menu":G=GET:CHAIN"MENU"
 170 END
 180 :
1000 DEFPROCsetup
1010 rec=1:FDR=256
1020 B%=HIMEM:maxf=12:maxs=6
1030 DIMrecord$(maxf),field$(maxf),width%(maxf),os 22
1040 DIMsfld%(maxs),strt%(maxs),len%(maxs)
1050 ENDPROC
1060 :
2000 DEFPROCtitle
2010 FORI=0TO1:
     PRINTTAB(3,I)CHR$141CHR$129CHR$157CHR$131"BEEBUG
     FILER SORT ROUTINE  "CHR$156:NEXT
2020 ENDPROC
2030 :
3000 DEFPROCinput
3010 REPEAT
3020 PROCwindow1:CLS:PROCwindow2
3030 PROCfilenames
3040 PROCsortfield
3050 IFsf=1PROCalfnum
3060 PROCascdec
3070 UNTILFNask("Confirm (Y/N): ")<3
3080 ENDPROC
3090 :
4000 DEFPROCfilenames
4010 REPEAT
4020 PRINTCHR$130"Enter name of file to be sorted:
     "'CHR$131;
4030 INPUTTAB(15)""op$:PROCopen(op$)
4040 UNTILF1:PRINT
4050 REPEAT
4060 PRINTCHR$130"Enter new name of sorted file   :
     "'CHR$131;
4070 INPUTTAB(15)""np$:PROCnopen(np$)
4080 UNTILF2:PRINT
4090 PROCwindow1:PRINTCHR$134"Old file:
     "CHR$131;op$TAB(20)CHR$134"New file:"CHR$131;np$
4100 ENDPROC
4110 :
```

....*FILERS* V B2.1 Contd>>

....*FILERS* V B2.1 Contd>>

```
5000 DEFPROCsortfield
5010 PRINTTAB(9,3)CHR$134"SORTFIELD COMPOSITION"TAB(0,5)
     CHR$133"Field"SPC6"Width  Start  Length  Total"
5020 PRINTTAB(32,6)CHR$134"<=";maxl
5030 lf=0:sf=0
5040 REPEAT:PROCwindow2:REPEAT
5050 PRINTCHR$130"Enter field name";:
     IFsf>0PRINT", or <RETURN>"'CHR$130
     "to end composition";
5060 PRINT":"CHR$131;:INPUT""sfld$
5070 UNTIL(sf>0ANDsfld$="")ORFNfwidth(sfld$)
5080 IFsfld$=""GOTO5250 ELSEsf=sf+1
5090 sfld%(sf)=R:strt%(sf)=1:len%(sf)=FNfwidth(sfld$)
5100 PROCwindow1:PRINTTAB(0,sf+6)CHR$131;
     sfld$TAB(16-LEN(STR$(len%(sf))))CHR$134;len%(sf):
     PROCwindow2
5110 REPEAT
5120 PRINTCHR$130"Enter start position in field or press"
     'CHR$130"<RETURN> for entire field:"CHR$131;
5130 INPUT""strt$:strt%=VALstrt$
5140 UNTILstrt$=""OR(strt%>0ANDstrt%<=len%(sf))
5150 IFstrt$<>""strt%(sf)=strt%
5160 PROCwindow1:
     PRINTTAB(23-LEN(STR$(strt%(sf))),sf+6)CHR$131;
     strt%(sf):PROCwindow2
5170 IFstrt$=""GOTO5230
5180 REPEAT
5190 PRINTCHR$130"Enter length of fieldblock or press"
     'CHR$130"<RETURN> for rest of field  :"CHR$131;
5200 INPUT""len$:len%=VALlen$
5210 UNTILlen$=""OR(len%>0ANDlen%+strt%(sf)<=len%(sf)+1)
5220 IFlen$<>""len%(sf)=len%
     ELSElen%(sf)=len%(sf)-strt%(sf)+1
5230 IFlf+len%(sf)>maxl VDU7:
     PRINTTAB(8)CHR$129"SORTFIELD TOO LONG":
     GOTO5180 ELSElf=lf+len%(sf)
5240 PROCwindow1:
     PRINTTAB(30-LEN(STR$(len%(sf))),sf+6)CHR$131;
     len%(sf)TAB(38-LEN(STR$(lf)),sf+6)CHR$134;lf
5250 UNTILsf=maxs ORsfld$="" ORlf=maxl
5260 ENDPROC
5270 :
```

....*FILERS* V B2.1 Contd>>

```
6000 DEFPROCalfnum
6010 PROCwindow2:
     PRINT'CHR$130"Press 'A' for alphabetic sort"'TAB(3)
     CHR$130"or 'N' for numeric sort :";
6020 REPEAT *FX15 1
6030 A=INSTR("AaNn",GET$):UNTILA
6040 PROCwindow1:PRINTTAB(11,9)CHR$131;
6050 IFA>2t$="VAL":PRINT"Numeric sort" ELSEt$="":
     PRINT"Alphabetic sort"
6060 ENDPROC
6070 :
7000 DEFPROCascdec
7010 PROCwindow2:
     PRINT'CHR$130"Press 'A' for ascending order"'TAB(3)
     CHR$130"or 'D' for descending order :";
7020 REPEAT *FX15 1
7030 A=INSTR("AaDd",GET$):UNTILA
7040 PROCwindow1:PRINTTAB(11,sf+9)CHR$131;
7050 IFA>2o$="<":PRINT"Descending order" ELSEo$=">":
     PRINT"Ascending order"
7060 ENDPROC
7070 :
8000 DEFPROCopen(p$):LOCALI
8010 IFp$=""VDU7:PRINTTAB(10)CHR$129"NO FILE GIVEN":
     F1=0:ENDPROC
8020 F1=OPENIN(p$)
8030 IFF1=0VDU7:PRINTTAB(10)CHR$129"NO SUCH FILE":ENDPROC
8040 PTR#F1=0:INPUT#F1,rec,recn,recs,f:
     maxl=(&7C00-B%)DIV(rec-1)-4
8050 FORI=1TOf:INPUT#F1,field$,width%(I):
     field$(I)=FNstrip(field$,"."):NEXT
8060 CLOSE#F1:ENDPROC
8070 :
9000 DEFPROCnopen(p$):LOCALI
9010 IFp$=""VDU7:PRINTTAB(10)CHR$129"NO FILE GIVEN":F2=0:
     ENDPROC
9020 F2=OPENIN(p$):CLOSE#F2
9030 IFF2<>0VDU7:PRINTTAB(10)CHR$129"FILE EXISTS":
     IFFNask("Overwrite existing file? (Y/N) : ")>2 F2=0:
     ENDPROC
9040 PROCoscli("SAVE "+p$+" 0+"+STR$~(FDR+recn*recs))
9050 F2=OPENUP(p$):PTR#F2=0:PRINT#F2,rec,recn,recs,f
9060 FORI=1TOf:PRINT#F2,LEFT$(field$(I)+
     STRING$(12,"."),12),width%(I):NEXT
9070 CLOSE#F2:ENDPROC
9080 :
```

....*FILERS* **V B2.1 Contd>>>**

....*FILERS* V B2.1 Contd>>

```
10000 DEFPROCclose:VDU7
10010 PRINTTAB(2)CHR$134"Files closed - ";rec-1;
      " records in use"
10020 CLOSE#0
10030 ENDPROC
10040 :
11000 DEFPROCsort
11010 LOCALI%,J%,K%,L%,U%,T$:L%=lf+4:PROCwindow2:
      PRINTCHR$133CHR$136;
11020 F1=OPENIN(op$):F2=OPENUP(np$)
11030 s$="":FORI%=1TOsf:
      s$=s$+"MID$(record$("+STR$sfld%(I%)+"),"+STR$strt%
      (I%)+","+STR$len%(I%)+")+":NEXT:s$=LEFT$(s$,LENs$-1)
11040 PRINTTAB(13,0)"READING"
11050 FORI%=1TOrec-1:PROCread(I%)
11060 $(B%+L%*(I%-1))=EVALs$+RIGHT$("00"+STR$I%,3):NEXT
11070 t$=t$+"T$"+o$+t$+"$(B%+L%*(I%-1))"
11080 PRINTTAB(13,0)"SORTING"
11090 K%=1:REPEAT K%=3*K%+1:UNTIL3*K%+1>=rec-1
11100 REPEAT K%=K%DIV3
11110 FORJ%=K%+1TOrec-1:T$=$(B%+L%*(J%-1)):I%=J%-K%
11120 REPEAT U%=EVALt$:
      IFNOTU% $(B%+L%*(I%+K%-1))=$(B%+L%*(I%-1)):I%=I%-K%
11130 UNTILU%ORI%<1:$(B%+L%*(I%+K%-1))=T$
11140 NEXT:UNTILK%=1
11150 PRINTTAB(13,0)"WRITING"
11160 FORI%=1TOrec-1:J%=VALRIGHT$($(B%+L%*(I%-1)),3)
11170 PROCread(J%):PROCwrite(I%)
11180 NEXT:CLS:ENDPROC
11190 :
13000 DEFFNask(msg$)
13010 LOCALA:PRINTCHR$134msg$;CHR$131;
13020 REPEAT:A=INSTR("YyNn",GET$):UNTILA:
      IFA>2THENPRINT"N"ELSEPRINT"Y"
13030 =A
13040 :
14000 DEFPROCwindow1
14010 VDU28,0,21,39,3:ENDPROC
15000 DEFPROCwindow2
15010 VDU28,0,24,39,21,12:ENDPROC
15020 :
16000 DEFPROCread(n):LOCALI
16010 PTR#F1=FDR+recs*(n-1)
16020 FORI=1TOf:INPUT#F1,record$(I):NEXT
16030 ENDPROC
16040 :
```

....*FILERS* V B2.1 Contd>>>

....*FILERS* V B2.1 Contd>>>

```
17000 DEFPROCwrite(n):LOCALI
17010 PTR#F2=FDR+recs*(n-1)
17020 FORI=1TOf:PRINT#F2,record$(I):NEXT
17030 ENDPROC
17040 :
18000 DEFFNstrip(p$,c$):LOCALI:I=LENp$+1
18010 REPEAT:I=I-1:UNTILMID$(p$,I,1)<>c$ORI=0
18020 =LEFT$(p$,I)
18030 :
19000 DEFFNfwidth(p$):IFp$=""=0
19010 R=0:REPEAT:R=R+1:UNTILp$=field$(R)OR R=f
19020 IFR=f ANDp$<>field$(R)VDU7:PRINTTAB(10)CHR$129
      "NO SUCH FIELD"':=0 ELSE =width%(R)
19030 :
20000 DEFPROCoscli($os)
20010 LOCALX%,Y%
20020 X%=os:Y%=os DIV256:CALL&FFF7
20030 ENDPROC
20040 :
21000 DEFPROCerror
21010 IFERR=17PRINT'TAB(12)CHR$129"SORTING ABORTED":
      GOTO21040
21020 ONERROROFF
21030 REPORT:PRINT" at ";ERL
21040 PROCclose:CHAIN"MENU"
21050 ENDPROC
21060 :
```

Beeblet acknowledges with gratitude the permission granted by *Beebug* magazine to publish the series of articles on *Beebug* FILER. For previous articlcs, see *Beeblet* Vol 4 Nos 8, 9 and 10; Vol 5 Nos 1 to 5 inclusive.

FILER NOTES

A series of Notes are also available summarising the features and commands of FILER. These are in the form of VIEW files. There is no problem reading these files directly into INTERWORD and spooled versions could be made available. If you would like copies of these notes, just add an appropriate request when you send in your next contribution to *Beeblet*.

Discovered on a

BULLETIN BOARD

Glossary of Terms and Quotes

from "The Official Computer Hater's Handbook"
by D.J. Arneson (Dell Publ. Co. 1983)

BYTE	Short for "buy it". Refers to how many peripherals you'll have to purchase to support a computer (e.g. there are 8 "buy it", 16 "buy it", or 64 "buy it" computers)
CARRIAGE RETURN	What you do if you decide not to keep the kid.
COLD BOOT	What a programmer puts on feet in winter.
COMPUTER SCIENCE	The fastest growing voodoo art course in American colleges and universities today.
DATA	A nice-a Italian girl.
DOWN	Said of a computer made of feathers: "That computer is down."
DUMP	The best place for computers
ELECTRONIC MAIL	New Zealand Post jargon for anything delivered in less than a week.
EXPANSION SLOTS	The extra holes in your belt buckle.
FLOPPY DISK	Serious curvature of the spine.
FRIENDLY	Said of anything associated with computers that is difficult to understand or does not work the way it says it will.
INTELLIGENT	A hypothetical term used in computing.
INTERLACE	To tie two boots together.

INTERPRETER	The person you take with you to the computer store to help you understand what the salesman is talking about; usually a twelve year old kid.
KEYPAD	An apartment with a lock.
KEYWORD	"Your place or mine?"
LIGHT PEN	A minimum sentence prison.
LINE FEED	"I've never met anyone as interesting as you before," etc.
MENU	An itemised list of ways to make a mistake on a computer.
MULTI-PROCESSING	To cook statistics more than once.
NETWORK	What fishermen do when not fishing.
OUTPUT DEVICE	A word processor who can't say "No!"
PERIPHERAL	Anything that costs a lot of money that can remotely be associated with computers.
PROGRAM	A random accumulation of bugs.
PROGRAM COUNTER	A parent who limits how much TV you watch.
RAM	Where most of the bugs are kept.
ROM	Where you put all the bugs that won't fit in the ram.
ROUTINE	A program that never works the same way twice.
SILICON	A foolish prisoner.
SOFTWARE	What hackers wear under their hardware.
SUPERFICHE	Moby Dick, Jaws, etc.
TERMINAL INTELLIGENCE	To be so smart it kills you.

TIME SHARING	What you do when only one of you has a watch.
VARIABLE	Anything with a fixed value you thought would change.
WORD PROCESSING	To change the meanings of words, e.g., to make the simplest communication completely unintelligible.
WORK STATION	A desk with a dumb terminal sitting at a computer.
ZAP	The ultimate bug killer.

HOORAY! You've mastered the lingo! Now, just remember:

Those who can, compute.
Those who cannot, program.
Those who can't program, write manuals.
Those who can't write manuals, sell computers!

::

Not very original?

Then what are your favourite definitions?

Remember "What is a Girl?"
Can anybody define "What is a BBC Computer User?"

Or Electron. Or Master. Or Archimedes.

PROGRAMMING RED KEYS

Richard Bayliff, Takaka, writes enquiring how to program the red function keys so that he can use <SHIFT>*fx*, <CTRL>*fx*, and even <SHIFT><CTRL>*fx*.

Since this involves the use of *FX calls, Anne Flower has offered to cover this subject in the next issue. Care is required as there are limitations that need to be understood; and there are ways round those limitations.

An interesting enquiry - should be an answer of interest to a number of readers. So watch for Anne's response in the next issue.

DATABASE SERVICES

COMMSTAR 2	Videotex/ASCII communications software for all models of BBC	$ 99.00
TELEPHONE ACCESSORIES	Jackpoints, dual adapters, extension leads, US to BT conversion cables from	$ 9.50
MODEMS	MD312 Manual dial, multispeed 300/300 1200/75 - Telecom approved	$ 375.00
	DISCOVERY 1200C Autodial multispeed 300/300, 1200/1200 - Telecom approved	$ 759.00

Contact:

DATABASE SERVICES

PO BOX 94, AUCKLAND.
STARNET: 9:DRV001

11 ANZAC AVE., AUCKLAND
Telephone: (09) 799658

~~~~~~

For Sale:

VIEW ROM: Still in box with Manuals, Keystrip, etc. Ideal for BBC B $150

SIDEWAYS RAM 16K (Watfords): Uses 2x8K RAM chips on carrier board with flying lead. Plugs into spare ROM slot. Comes complete with instructions and write protect switch, utility software for loading, saving. + printer buffer. (note: Same as ACORN unit) $ 55

Contact: Frank Pinion .. phone (03) 519 219

****

MEMBERSHIP

Membership of the National User Group is on payment of an annual subscription from April to March, which is $20.00 for the 1988/9 year. A local branch levy may be payable which entitles attendance at meetings as detailed on the back page. Joining or renewing members get all the back issues of *Beeblet* for the current year - those joining after 1 October may have half the annual subscription waived and only receive *Beeblet* from 1 October.

****
~~~~~~

USER GROUP MEETINGS

AUCKLAND: -2nd and 3rd Wednesday of each month at Social Club, Auckland Hospital, Grafton Road at 7.30pm. Contact John Buchanan (Secy), phone 546-985; David Wilkin (Pres), phone 437-540.

North Shore: -last Wednesday of the month, 7.30pm, Salvation Army Hall, Glenfield Road, Glenfield. Contact: Chris, Phone 419-0543 (wk).

CHRISTCHURCH: -fortnightly workshops, Monday 6.30pm at Hagley High School. phone Michael 582-267.

DUNEDIN: -first Thursday of the month. Contact: Martin McDowall, Phone 42-031

HAWKES BAY: -2nd Sunday 2.30pm, Taradale Primary School. Contact Ron: 444-970, John: 445-729 (librarian), Major: 447-729, Mitch: 85-395.

INVERCARGILL: -Southland Education Centre, 70 Doon St. Contact Bob Evans, 73-050 or Lyndon McEntee (020) 87-365.

NELSON: -second Monday of the month, 7.30pm, Nelson Baptist Church, Bridge St. Contact: Peter, Richmond 8975; Ken, Nelson 85-150; Richard, Richmond 8489.

PALMERSTON NORTH: -Phone Brian Knowles 79-319.

ROTORUA: -first Tuesday of month, Science Lab at Rotorua Girls High School. Phone: Ian 85-583, Joan 81-536 Tony 89-316.

TAUPO: -every second Wednesday, 7.00pm. Contact: David 84-215 (h), 85-124 (wk).

TAURANGA: -every second Wednesday. ph. Chris & Jane 65-076.

TIMARU: -Phone Lloyd van der Krogt Timaru 61-412.

TOKOROA: -1st Friday of month, 7.30pm, Tokoroa High School. Contacts: Don Fraser, 68-813 or Graeme Robertson, 65-346 Tokoroa.

WAIKATO: -first Monday of the month, 7.30pm, Waikato Tech Institute. Ph. Alison or John, Morrinsville 6695.

WELLINGTON: -usually second Wednesday, 7.30pm, Royal Society Lecture Theatre, Turnbull St, Thorndon.
-Workshops: fourth Tuesday of month, other details: ph. Don 848-235 or Mike 785-437.

OTHER CENTRES: -let us know and we will publish details here.

BEEBLET

PUBLISHED: monthly except January and mailed to financial members.

CONTRIBUTIONS: -most welcome. Listings should be sent in on tape or disc. (Tape - 2 copies, one at 300 baud please). Please include written explanation of listing; text and graphics only material welcome too.

DEADLINES: -For a particular issue, the last day for material is the fifteenth day of the month prior to publication.

ADVERTISING: -No fixed charge is made for advertising. Supply camera ready, deadlines as above.

Copyright (c) BBC/ACORN COMPUTER USER GROUP OF NZ INC.

Registered Publication with NZPost

THE BBC/ACORN COMPUTER USER GROUP OF NZ INC, PO Box 9592, WELLINGTON

Reproduced by SANDIFORD PRINTERS